|  |
| --- |
| **DACS 2101 - Discrete Structures** |
| Homework #2 – Fall 2024 |

**Here are the guidelines for your upcoming homework, which is due on Tuesday, October 29th:**

* **Handwritten Answers:** Please submit handwritten answers, **not typed**.
* Utilize A4 papers for physical copy submissions.
* Include all necessary details on the cover page, such as your name, course title, and instructor's name.
* Present your work in a clear and comprehensible manner.
* Explicitly mention the question number for each response (e.g., question number and sub-number).
* Ensure that your assignment pages are numbered.
* Kindly use a stapler to secure your papers before submission.
* **Hard Copy Submission:** Bring a hard copy of your homework to the lecture on**Tuesday, October 29th.**
* **Soft Copy:** Additionally, you need to submit a soft copy of your homework **PDF type ONLY** on Dropbox, in addition to the circuit file and the python code. **Please note that the hard copy will be graded for everything else.**
* **Late Submission:**Submissions late by up to three days will receive a 10% penalty per day. No submissions overdue by more than three days will not be accepted, unless extenuating circumstances (e.g., documented illness) are provided.
* **Academic Integrity:** Students are expected not to copy each other's answers. Any instances of copying may result in penalties.

Please ensure you adhere to these guidelines for a smooth and fair homework submission process. If you have any questions or need clarification, do not hesitate to reach out.

**Part 1: Boolean Algebra:**

**1.** Write x ⊕ y in disjunctive normal form and conjunctive normal form. Using of these forms, simplify these expressions: [2 marks]

a) x ⊕ 0

b) x ⊕ 1

c) x ⊕ x

d)

**2.** Find the sum-of-products expansions of the Boolean function F(x, y, z) that equals 1 if and only if: [2 marks]

a) x = 0.

b) xy = 0.

c) x + y = 0.

d) xyz = 0

**3.** Express each of these Boolean functions using the operators ⋅ and −. [4 marks]

a) x + y + z

b)

c)

d)

**Part 2: Sets:**

**1.** For each of these pairs of sets, determine whether the first is a subset of the second, the second is a subset of the first, or neither is a subset of the other. [2 marks]

**a)** the set of people who speak English, the set of people who speak English with an Australian accent

**b)** the set of fruits, the set of citrus fruits

**c)** the set of students studying discrete mathematics, the set of students studying data structures

**d)** the set of students in the college of computing and IT, the set of students at UDST

**2.** Determine whether these statements are true or false. [4 marks]

**a)** ∅ ∈ {∅}

**b)** ∅ ∈ {∅*,* {∅}}

**c)** {∅} ∈ {∅}

**d)** {∅} ∈ {{∅}}

**e)** {∅} *⊂* {∅*,* {∅}}

**f )** {{∅}} *⊂* {∅*,* {∅}}

**g)** {{∅}} *⊂* {∅*,* {{∅}}}

**h)** ∅ *⊂* {{∅}*,* {{∅}}}

**3.** Can you conclude (with a proof) that *A* = *B* if *A*, *B*, and *C* are sets such that [9 marks]

**a)** *A* ∪ *C* = *B* ∪ *C*?

**b)** *A* ∩ *C* = *B* ∩ *C*?

**c)** *A* ∪ *C* = *B* ∪ *C* and *A* ∩ *C* = *B* ∩ *C*?

**4.** Sets in python [7 marks]

For the following questions you have to provide the code and the output of your code.

**a)** Write a code to generate the set setA of positive integers less than 1000 which last two digits are divisible by 43 (example 186 is in the set because 86 is divisible by 43)

**b)** Write a code to generate the set setB of positive integers less than 1000 which first two digits are divisible by 24 (example 725 is in the set because 72 is divisible by 24)

**c)** Write a code to compute setC the union of setA and setB

**d)** Write a code to compute setD the intersection of setA and setB

**e)** Check using a python code that |setA|+|setB|=|setC|+|setD|

**f)** Write a function cartesian that has two sets as input and returns their cartesian product as output (without importing any library)!

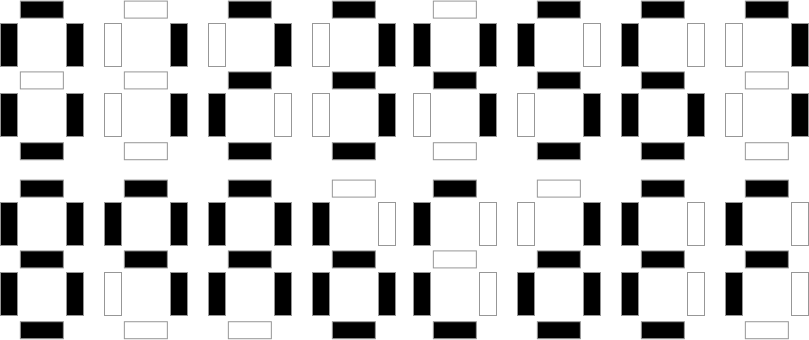
**g)** Use your function to find the cartesian product between setD and the set of colors on a traffic light.

**Part 3: Design of a combinational circuit for a 7-segment display**

[20 marks]

The aim of this part is to show a concrete application of combinational circuits by making you design a (somehow) useful circuit controlling a simple electronic component called a 7-segment display.

A 7-segment LED display is a classical output device for clocks and other small displays. It can be used to display numbers and even letters as shown below.



We denote the seven segments of the display s1 to s7 as follows:

s1

s2

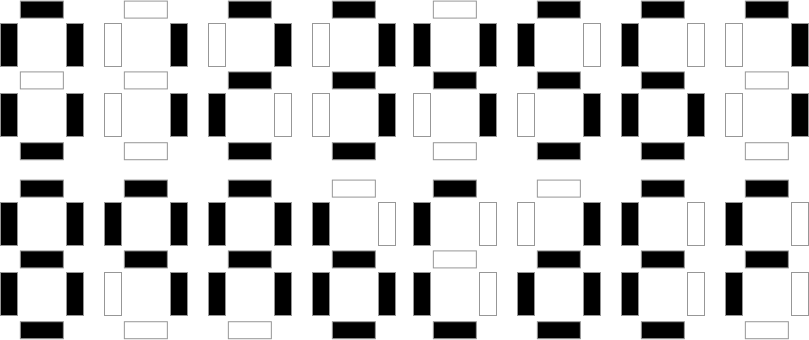
s5

s6

s7

s4

s3



The display works as follows: if the segment should be white, its value is 0 otherwise (the segment should be black) its value is 1.

**Your task is to design a circuit with three bits input i2, i1 and i0 (to represent numbers from 0 to 7) that will display your student ID. More precisely, it will output the digit of your student ID at the position corresponding to the input.**

**For example**, assume that your student ID is 61287654. Then the 7-segment display should show:

* 4 (which means only s3, s5 and s6 are 0 all the others are 1) if the input is 0,
* 5 if the input is 1,
* 6 if the input is 2,
* 7 if the input is 3,
* 8 if the input is 4,
* 2 if the input is 5,
* 1 if the input is 6,
* 6 if the input is 7

Of course, the input is in binary but the conversion is very easy to do. For example the input is 4 means 100 which means i2=1 and i1=i0=0.

This means that you will have a circuit with three bits input and seven bit outputs (one for each segment).

The circuit will be different for each student because you all have different student IDs J

**Show all the steps that you use to design the circuit, including truth table and minimization using K-maps.**

After generating the Boolean expressions of s1, s2…s7 you have to draw the circuit.

To draw a nice circuit, use the software logisim. Logisim is a free software (you can download it from <https://sourceforge.net/projects/circuit/?source=navbar>

It is very intuitive to use and I advise you to go through the tutorial in the help section to understand the basic. For now you will only use the gates that we have covered in class and a seven segment display. The tool covers much more and if you are interested, you will have plenty of time to learn more about it in your free time.

The advantage of using the tool is that it will produce a neat circuit design and very easy to test (which is a big advantage both for you and me).

**What you have to deliver:**

* A report explaining how you designed the circuit (with your student ID, the truth table, minimization using K-maps).
* A .circ file which contains your circuit design in logisim.

As an example we provide the HW2.circ file which simply show the 7-segments display and how it is wired, and HW2\_example.circ which shows a simple circuit controlling two of the segments only.